



Excel UserForm 課題の自動採点システムの試作

田 中 敬 一

概要 Excel はオフィスアプリケーションの中で最も利用されているソフトウェアであり、表計算ソフトとしてデータ集計作業の多くの処理に活用されている。そのため、データ集計作業の効率化は重要なテーマである。Excel を使った効率的な作業の自動化は Excel VBA を導入することで可能となる。Excel VBA は UserForm を使ったインターフェースが作成でき、より簡単にデータ入力することや、視覚的に分かりやすい画面が利用できる。本稿では UserForm を学習させるための教材を作成し、設定された UserForm 課題の自動採点処理システムについて報告するものである。

キーワード Excel VBA, UserForm, 自動採点システム, 情報リテラシー教育

原稿受理日 2010年1月15日

Abstract Excel is the most widely used MS Office application. As a spreadsheet, it is used in most data aggregation processing. For that reason, an important theme is the efficiency of data aggregation processing. It is possible to use Excel VBA to automate the efficient processing inside Excel. With Excel VBA, it is possible to create an interface using UserForms, which results in easier data input and easy-to-visualize forms/screens. Teaching material was prepared for learning UserForms. This paper reports on the UserForm task set for an automatic rating processing system.

Key words Excel VBA, UserForm, Automatic rating system, Information literacy education

1. はじめに

情報技術の発展にともなって多くのコンピュータが利用される時代となり、いたるところで IT スキルが要求されている。企業活動においても同様で、多様な業務にコンピュータが利用されている。このため、大学におけるコンピュータ教育は日に日に重要性が増しており、文系学部においても、多くの大学でコンピュータ教育のカリキュラムが大幅に強化され導入されている。

近畿大学経済学部は平成20年度カリキュラムより大幅な IT 教育の見直しを行った。これは時代の要請に応えるよう、基礎的な内容から応用的な内容まで体系的に学習できる開講科目を準備し、IT 知識の修得ができるカリキュラムを導入したものである。これらのカリキュラムには、パソコンの操作を学習するものや IT に関する知識を得るものやその他関連した内容を学ぶものなど多彩な科目群となっている。パソコン操作の学習はオフィス系アプリケーションとしてワープロ、表計算、プレゼンテーション、データベースなどの実習を行っているが、特に表計算 Excel の修得に力を入れている。表計算ソフトはデータ集計やデータ分析やグラフ作成など、いろいろな処理に利用することが可能である。データ集計作業などの事務作業には必須のツールとなっているため、経済学部では Excel を使った効率的な作業の自動化処理は重要と考えている。Excel の自動化処理には Excel VBA (Visual Basic for Applications) の知識が必要であるため、Excel VBA の学習カリキュラムも導入している。Excel VBA はプログラム言語の一種であるため、取得が容易ではないが、経済学部生全員が受講できる体制を構築している。

既報⁴⁾では Excel VBA を学習する学習教材を作成し、コンピュータによる自動採点システムを報告しているが、本論は UserForm を学習させるための教材を作成し、設定された UserForm 課題の自動採点処理システムについて報告するものである。

2. Excel 授業の取り組み

経済学部における Excel 操作を修得する開講科目は、表1のように1年後期から2年生前後期の授業カリキュラムに導入している。平成21年度の各科目の受講者数は750名程度であり、経済学部生のほぼ全員が履修している状況である。

1年後期配当のコンピュータ実習Ⅱは、Excel 入門としての位置づけの科目であり、初

めて Excel を操作する学生を対象としたカリキュラムとなっている。Excel の基本的な操作から簡単な関数機能の紹介やグラフやデータベース機能の学習を行っている。また、基礎的な知識の取得を目標にしているため、Microsoft Certified Application Specialist (MCAS) の資格取得を推奨しており、多くの学生が受験している状況である。表 2 には平成18年度から実施している経済学部 MOS/MCAS 試験の実施状況を示したものである。このうち、平成21年 2 月 9～10日に実施した MOS/MCAS 試験は、平成20年度入学の1年生が多く受験しており、科目別の合格者数を示したものが図 1 のグラフである。1年後期のコンピュータ実習Ⅱの Excel 入門の授業を受講した後、MCAS 試験 (Excel 2007) を受験して合格した人数が283名であることを示している。

2年前期のコンピュータ特修実習Ⅰは、1年後期に学習した内容を十分に活用できるように、実践的な関数や機能が学習できるカリキュラムとなっており、Excel の機能を十分に活用できる能力を身につけるものとなっている。また、平成21年の7月にはコンピュータ特修実習Ⅰの受講者を対象とした日本商工会議所主催の日商 PC 検定 (データ処理) 3 級を学内施設で実施し、27名が受験し12名が合格した。

2年後期のコンピュータ特修実習Ⅱは、1年後期から1年間に渡って学習した Excel の機能を前提として、Excel VBA を学習する内容となっている。この授業は経済学部生に

表 1 Excel 操作を修得する開講科目

配当学年	科目名	内容	受講者数 (平成21年度)
1 年後期	コンピュータ実習Ⅱ	Excel 入門	756名
2 年前期	コンピュータ特修実習Ⅰ	Excel 応用	739名
2 年後期	コンピュータ特修実習Ⅱ	Excel VBA 入門	735名

表 2 経済学部 MOS/MCAS 試験実施状況

実施日	申込	受験	合格	不合格	欠席	合格率
H18. 8. 2～3	335	320	287	33	15	89.7%
H19. 2.15～16	472	450	380	70	22	84.4%
H19. 8. 2～3	441	422	366	56	19	86.7%
H20. 2.15～16	444	424	368	56	20	86.7%
H20. 8. 4～5	493	473	422	51	20	89.2%
H21. 2. 9～10	501	478	448	30	23	93.7%
H21. 8. 3～4	678	647	605	42	31	93.5%

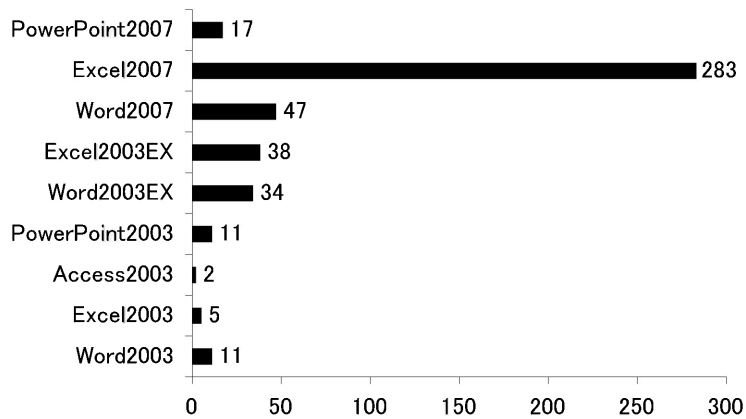


図1 経済学部 MOS/MCAS 科目別合格者数 (H20年度後期)

とって初めて学習する本格的なプログラム言語の科目となる。Excel を題材としたプログラム言語であるため、一般的なプログラム言語よりは理解しやすい傾向があると言える。しかし、プログラム言語であるのには変わりがないので、落ちこぼれが発生しないような授業の進行が要求される。このような状況により、この授業でも目標資格である Excel VBA エキスパート試験 (Basic) の学内試験を開催し、Excel VBA の学習に役立っている。現在、学内で実施している Excel VBA エキスパート試験 (Basic) の出題項目を表3⁽⁷⁾に示す。

コンピュータ特修実習Ⅰ・Ⅱのカリキュラム内容を示したものが表4である。このように、経済学部では Excel の実習授業を多く開講して、VBA も含めた Excel の操作知識を修得させている。本論で試作した自動採点システムはコンピュータ特修実習Ⅱで学習する Excel VBA の教材の一つとして考えているが、Excel VBA エキスパート試験 (Basic) には UserForm の操作が出題範囲とはなっていない。しかし、Excel VBA を学習するにあたっては、UserForm の学習が重要と考えているため、本論でそのシステムを開発してみることにする。

表3 ExcelVBA エキスパート試験の出題範囲（試験科目：Excel VBA Basic）

大 分 類	小 分 類
1 マクロと VBA の概念	マクロとは何か VBA とは何か VBA の特徴 Visual Basic Editor の操作と画面 ブックとマクロの関係 セキュリティレベル
2 マクロ記録	マクロ記録とは マクロ記録の限界 マクロ記録の活用方法 記録のしかた 記録される場所 標準モジュール 個人用マクロブック
3 VBA の構文	オブジェクト式 ステートメント 関数 演算子 モジュールとプロシージャ
4 変数と定数	変数の意味 変数の宣言 変数の代入と取得 スコープ 定数の意味と使い方
5 セルの操作	Range と Cells セルにデータを入力する セルのデータを取得する セルの書式設定
6 ステートメント	If ~ End If For ~ Next With ~ End With
7 ブックの操作	ブックを開く ブックを保存する ブックを閉じる
8 シートの操作	シートを挿入する シートを削除する シートに関連する基本的な操作
9 デバッグの基礎	デバッグとは 論理エラーと文法エラー イミディエイトウィンドウ ブレークポイント ステップ実行
10 マクロの実行	Visual Basic Editor から実行する [マクロ] ダイアログボックス シート上にボタンを配置する メニュー/ツールバーに登録する クイックアクセスツールバーに登録する

表4 平成21年度授業項目および内容

科目名	コンピュータ特修実習Ⅰ	コンピュータ特修実習Ⅱ
配当学年	2年前期	2年後期
第1回	MCAS Excel の復習 (1)	Excel VBA の基礎知識, マクロ記録 (1)
第2回	MCAS Excel の復習 (2)	マクロの登録, マクロ記録 (2)
第3回	関数機能 (1)	VBE の操作, マクロ記録 (3)
第4回	関数機能 (2)	VBA の基本用語 (1), マクロ記録 (4)
第5回	関数機能 (3)	VBA の基本用語 (2), ヘルプの操作
第6回	関数機能 (4)	ブックとシート
第7回	関数機能 (5)	セル操作 (1)
第8回	データベース機能 (1)	セル操作 (2), データベース操作
第9回	データベース機能 (2)	変数と配列, ステートメントの操作 (1)
第10回	データベース機能 (3)	ステートメントの操作 (2)
第11回	総合問題 (1)	マクロ課題の作成 (1)
第12回	総合問題 (2)	マクロ課題の作成 (2)
第13回	総合問題 (3)	VBA エキスパート試験, ユーザフォームの操作
第14回	定期試験	定期試験

3. システムの概要

今回開発したシステムは、Excel VBA の学習システムとして Excel VBA 上で作成する UserForm の課題を対象とした自動採点システムである。開発したシステム環境は表5のとおりである。

表5 開発システムの環境

OS	Windows Vista
Office Application	Microsoft Excel 2007
統合開発環境	Microsoft Visual Studio 2005
プログラミング言語	Microsoft Visual Basic 2005

Excel VBA のプログラムは、汎用的なプログラムを作成する場合、標準モジュール上にマクロを作成するが、ユーザインターフェースをワークシート上のセルではなくフォーム上に作成するには、VBA プロジェクト上に UserForm を作成してプログラムを構築し

ていくことになる。Excel VBA は一般的にセル操作を対象としたデータの入出力が多い。しかし、一般的な Windows アプリケーションは UserForm を入力インターフェースとして用いることで、操作性の良いシステム開発が要求される。このような観点を考慮すると、Excel VBA の学習では単に標準モジュールに VBA プログラムをつくるのではなく、UserForm を使ったプログラミング処理の学習も重要であると考えられる。本システムは、この UserForm を学習するにあたって、学習効果を測定するための自動採点処理システムである。

図 2 には本システム処理のフローチャートを示している。また、図 3 には、Excel VBA の模擬問題が起動し、課題が表示され解答する時の画面である。

基本的な処理の流れとしては、Excel 起動後、制限時間チェック用のタイマーを起動させる。タイマーにより制限時間内であるかどうかを常時監視しており、時間に到達した場合には、総合採点結果を表示しプログラムを終了する。

制限時間内の場合には、あらかじめ準備している Excel の問題ファイルを問題 1 から順に開き、さらに、画面下部にはテキストデータとして準備している問題文を表示する。ここで解答者は、UserForm の問題を解答するため、VBE (Visual Basic Editor) を起動させ、UserForm を新規に挿入する。問題の内容に従って、コマンドボタンやテキストボックス、ラベル等のコントロールを追加する。さらにコマンドボタンをクリックなどのイベント処理プログラムを追加する場合には、そのコードをコードウィンドウに記述して設問を完成させていく。[リセット] ボタンは問題を最初からやり直す時に利用するため、このボタンをクリックした場合には再度問題を読み込み直す処理を行う。[次へ] ボタンをクリックした場合には、採点処理を実行した後、採点結果の表示を行い、次の問題の読み込みを行う。[終了] ボタンをクリックした場合には、採点処理につづき採点結果を示し、最後に総合採点結果を表示するプログラムとなっている。

図 4 は UserForm 問題の解答処理時の画面である。UserForm を作成するため、VBE を起動させ、UserForm を新規挿入で作成しコマンドボタンを配置している。また、コマンドボタンをクリックした場合の処理として、コードウィンドウに VBA プログラムを記載している部分を示している。なお、ワークシート上のボタン（フォームコントロール）[ボタン 1] はあらかじめ Excel VBA 模擬問題上に登録しており、UserForm が表示されるマクロを設定している【UserForm1.Show (vbModeless)】。

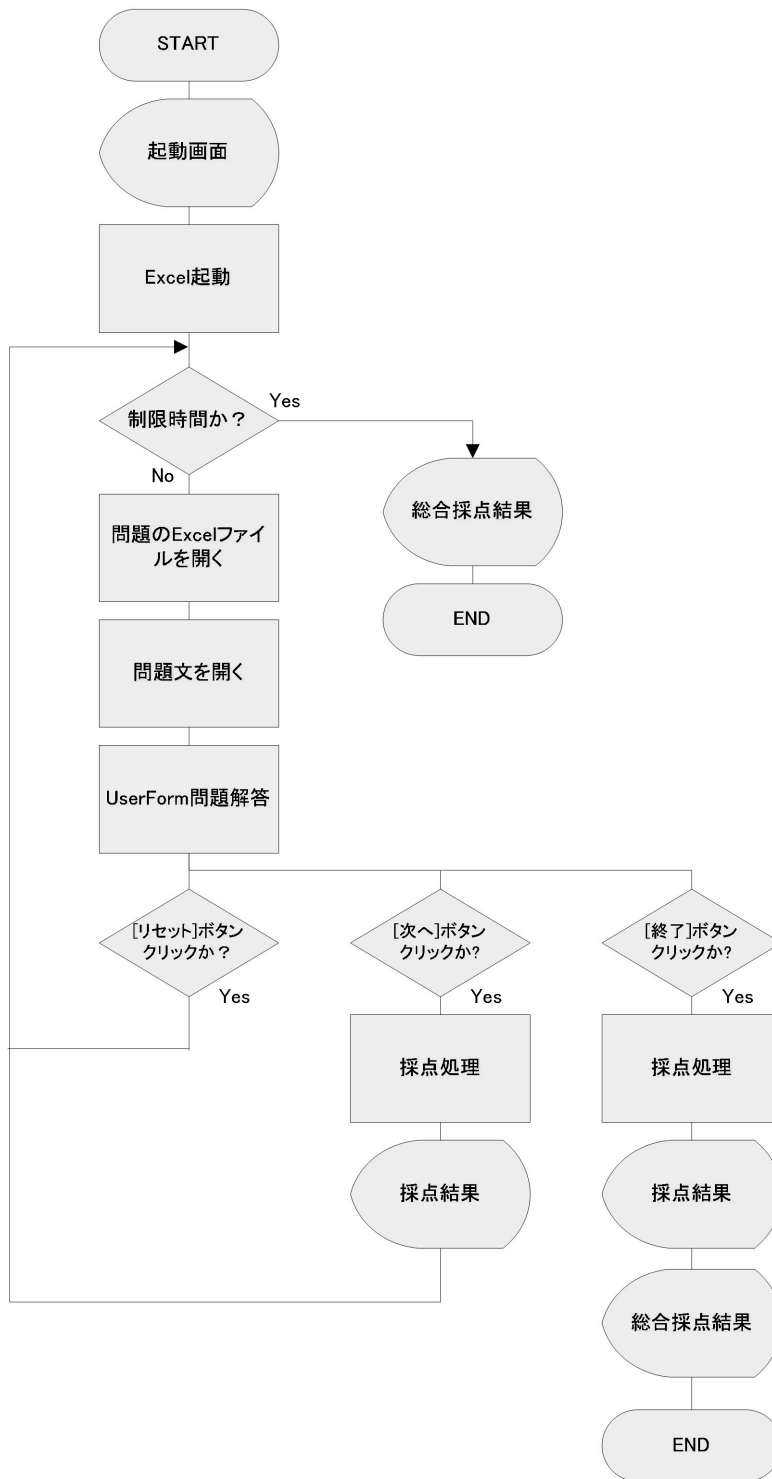


図2 システム概要

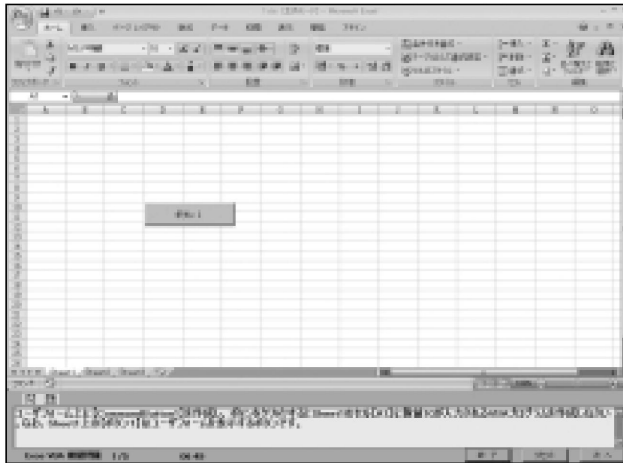


図3 Excel VBA 模擬問題起動画面

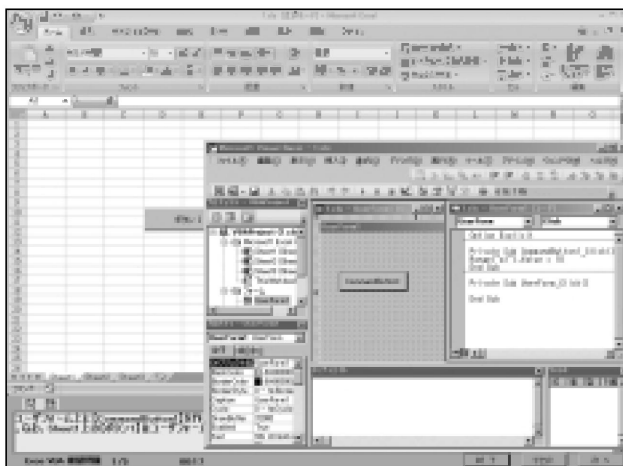


図4 VBA 模擬問題解答時の画面

4. 自動採点処理の概要

UserForm を組み入れた Excel VBA の問題において、採点処理を自動化するために次のような仕組みを検討した。

一般的なウィンドウアプリケーションはそのアプリケーションが実行されると独立したプロセスとして WindowsOS 上で扱われる。Excel も同様に、Excel が起動すると独立したプロセスとして稼働する。しかし、Excel 上で稼働する VBE ウィンドウや VBA 実

行時に表示される UserForm ウィンドウは独立したプロセスではなく、あくまで Excel の内部の子プロセスとして扱われるため、VBE や UserForm が起動している状態であってもプロセス数は増加しない。

今回の VBA 模擬問題は Visual Basic 2005 から Excel を起動したり、VBA のプログラムを実行したりすることはできるが、Visual Basic 2005 から Excel の子ウィンドウである UserForm を制御することは、難しい状況であることを確認した。特に UserForm 上にある各コントロールの部品を Visual Basic 2005 から制御するには各コントロールのウィンドウハンドルを取得する必要がある。一部のコントロールではハンドルを取得することもできるが、全てのコントロールに対してハンドル取得はできないことが判明した。そこで、UserForm のコントロールは Excel VBA 上から処理する仕組みとした。つまり、Excel の問題ファイルに制御用の Excel VBA プログラムを組み入れ、そのプログラムの実行を Visual Basic 2005 から制御する仕組みである。ただし、起動時に処理する内容の Excel VBA が標準モジュール上に前もって登録されていると、解答者にそのコードを参照されたり、また悪用されることが想定される。そこで、解答後の採点処理に入った段階で Visual Basic 2005 の制御により、標準モジュール上に Excel VBA プログラムのコードを組み入れることにした。Excel VBA 上から UserForm のコマンドボタンをクリックするためには、次のような仕組みを設けた。通常、UserForm のコマンドボタン処理は、解答者がコードウィンドウにプログラムを記述して解答している。そのときのコードは、【Private Sub CommandButton1_Click () ~End Sub】の間に記述することになる。ところが、後に標準モジュールに組み入れた UserForm の制御コードからコマンドボタンのクリックしたときのプロシージャを呼び出すには、【Private Sub CommandButton1_Click ()】では呼び出すことができない。【Private】を【Public】に書き換えた【Public Sub CommandButton1_Click () ~End Sub】でなければ外部から呼び出すことができない。そこで、Visual Basic 2005 の制御により UserForm のプログラムの置換を行っている。これにより、標準モジュールに組み入れ記載した macro10より UserForm 上のコマンドボタンのクリック処理が実現できる。

上記のように採点処理として Excel VBA プログラムコードの追加および修正を完了させた後、Visual Basic 2005 の制御によりワークシート上のボタン1をクリックし実行させる。そして、UserForm を表示させ、UserForm 上のコマンドボタンをクリックさせる。最後に UserForm を閉じて採点処理を終了している。その採点処理の結果をチェックし、正しく動作していれば「正解」とし、正しく動作していない場合には「不正解」と

して採点を行っている。なお、この UserForm コマンドボタンのクリック処理は、UserForm を作成していない場合、不完全な解答のため例外処理が発生し、エラーとして処理が止まってしまう場合がある。そのため、【Try～Catch～Finally】ステートメントを使い構造化例外処理を施している。図5は採点処理のシステムフローであり、表6に採点処理のサンプルコードを示している。また、表7は Excel VBA UserForm サンプル問題および解答例を示している。



図5 採点処理の概要

表6 採点処理サンプルコード (問題1)

```

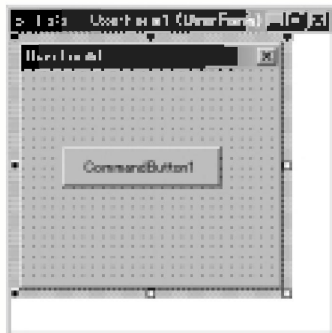
'標準モジュールの追加 -> add (1)
sMod = Form1.book.application.vbe.vbprojects (1).vbcomponents.add (1)
sCode = sMod.codemodule
'追加された標準モジュールに記載される
sCode.insertlines (2, "Sub macro10 ()")
sCode.insertlines (4, "UserForm1.Show (vbModeless)")
sCode.insertlines (5, "Call UserForm1.CommandButton1_Click")
sCode.insertlines (6, "Unload UserForm1")
sCode.insertlines (7, "End Sub")
Try
'CommandButton1 を Private から Public へ置換
cnt1 = Form1.book.application.vbe.vbprojects (1).vbcomponents ("UserForm1")._
codemodule.procbodyle ("CommandButton1_Click", 0)
Form1.book.application.vbe.vbprojects (1).vbcomponents ("UserForm1")._
codemodule.replaceline (cnt1, "Public Sub CommandButton1_Click ()")
'Worksheet のボタンに登録されているマクロを Macro1 から Macro10 に変更
Form1.sheet.Shapes (1).onaction = "macro10"
'Worksheet のボタン1 をクリックして実行
Form1.book.Application.run (Form1.sheet.Shapes (1).OnAction ())
'セル A1 の値を変数に代入
Areal = Form1.book.application.range ("a1").value
Catch ex As Exception
hantei = "不正解"
End Try
If Areal = 10 Then
hantei = "正解"
Else
hantei = "不正解"
End If

```

表7 Excel VBA UserForm サンプル問題と解答例

問題1 ユーザフォーム上に【CommandButton1】を作成し、ボタンをクリックすると Sheet 1のセル【A1】に数値10が入力される VBA プログラムを作成しなさい。なお、Sheet1上の【ボタン1】はユーザフォームを表示するボタンです。

解答例



5. お わ り に

近年の IT の進展により、ますます IT 処理能力の取得が重要となってきた。文系学部である経済学部生においても、PC を自由自在に使いこなせる能力が要求される状況となっている。このため、近畿大学経済学部では平成20年度カリキュラムより大幅な IT 教育の見直しを行った。この大幅な見直しの一つとして、表計算ソフトである Excel の学習強化を図った。Excel は非常に強力なツールではあるが、Excel VBA を使いこなせる能力を養うことで、さらに力を発揮させることができる。高度な情報活用能力を持たせるためには、Excel VBA が必須と考えている。そして、本学経済学部ではこの Excel VBA の授業をほぼ全員が履修するまでに至った。

本論は必要とされる Excel VBA 上で利用される UserForm の学習を行うため、UserForm に関連する問題についてコンピュータによる自動採点処理システムの構築を行い、Excel VBA を学習するための一提案を報告するものである。

今後の課題としては、実際の授業に適用し、コンピュータを使った自動採点処理と人による採点処理との差異を分析し、さらなる適切な評価ができる仕組みを検討してみたいと考えている。さらに、学習効率を上げるために、ネットワーク上に学習者の学習進捗状況を表示し、学習者同士が競争しあえる仕組みを構築したいと考えている。また、自動採点処理において、Excel VBA 上からコマンドボタン等のコントロールを行い、その結果を評価したが、Microsoft Active Accessibility⁽⁸⁾ を利用しながら Visual Basic 2005 から直接制御できる仕組みを検討課題としたい。

参 考 文 献

- (1) 井川はるき, 池谷京子, 2004. 『実践ワークショップ Excel VBA による Office 連携プログラミング』秀和システム.
- (2) 大村あつし, 2009. 『かんたんプログラミング Excel2007VBA [基礎編]』技術評論社.
- (3) ㈱オフィスエム, 2002. 『図解標準最新 VisualBasic.NET ハンドブック』秀和システム.
- (4) 田中敬一, 2008. 「VBA 学習システムの試作」『商経論叢』第55巻第1号, 329~335.
- (5) 田中亨, 2009. 『VBA エキスパート公式テキスト Excel VBA ベーシック』株式会社オデッセイコミュニケーションズ.
- (6) 富士通エフ・オー・エム株式会社, 2008. 『Microsoft Certified Application Specialist Microsoft Office Excel2007 公認テキスト&問題集』FOM 出版.
- (7) http://vbae.odyssey-com.co.jp/vbae/ex_b.html
- (8) [http://msdn.microsoft.com/en-us/library/ms697707 \(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ms697707 (VS.85).aspx)